

[illegible][illegible]

```

LL          IIIIII          SSSSSSSS
LL          IIIIII          SSSSSSSS
LL          II             SS
LL          II             SS
LL          II             SS
LL          II             SS
LL          II             SSSSSS
LL          II             SSSSSS
LL          II             SS
LL          II             SS
LL          II             SS
LL          II             SS
LLLLLLLLLLLL IIIIII          SSSSSSSS
LLLLLLLLLLLL IIIIII          SSSSSSSS

```

LPJ



F 6  
16-Sep-1984 00:05:20  
5-Sep-1984 14:00:08

VAX-11 FORTRAN V3.4-56  
DISK\$VMSMASTER:[ERF.SRC]LOGMSCP.FOR;1

Page 1

```
0001 c
0002 c Version: 'V04-000'
0003 c
0004 c*****
0005 c*
0006 c* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0007 c* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0008 c* ALL RIGHTS RESERVED.
0009 c*
0010 c* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0011 c* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0012 c* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0013 c* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0014 c* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0015 c* TRANSFERRED.
0016 c*
0017 c* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0018 c* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0019 c* CORPORATION.
0020 c*
0021 c* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0022 c* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0023 c*
0024 c*
0025 c*****
0026 c
0027 c
0028 c
0029 c Author Brian Porter Creation date 03-AUG-1983
0030 c
0031 c
0032 c++
0033 c Functional description:
0034 c
0035 c Modified by:
0036 c
0037 c V03-004 EAD0197 Elliott A. Drayton 21-Jun-1984
0038 c Changed code to use LSTLUN and RECORD_SIZE from SYECOM.
0039 c
0040 c V03-003 SAR0278 Sharon A. Reynolds 21-Jun-1984
0041 c Added TMSCP support.
0042 c
0043 c V03-002 SAR0171 Sharon A. Reynolds, 11-Nov-1983
0044 c - Removed the 'b_logmscp' routine.
0045 c - Changed the carriage control for use with ERF.
0046 c
0047 c v03-001 BP0001 Brian Porter, 07-SEP-1983
0048 c Added 'no-unit' code.
0049 c**
0050 c--
0051 c
0052 c
0053 c Subroutine LOGMSCP (lun,record_length)
0054 c
0055 c
0056 c include 'src$:msghdr.for/nolist'
0115 c include 'src$:emblmdef.for/nolist'
```

```
0184      include      'src$:syecom.for/nolist'
0312
0313      byte          lun
0314      integer*4     record_length
0315
0316      integer*2     logmscp_type
0317      equivalence   (emb(16),logmscp_type)
0318
0319      byte          mscp_packet(476)
0320      equivalence   (emb(36),mscp_packet(1))
0321
0322      integer*4     mscp$$b_opcode
0323      integer*4     mscp$$b_endcode
0324      integer*4     mslg$$b_format
0325      integer*4     packet_length
0326      integer*4     compress4
0327
0328
0329      call FRCTOF (lstlun)
0330      call HEADER (lstlun)
0331      call LOGGER (lstlun,'ERL$LOGMSCP ENTRY')
0332
0333      call LOGMSCP_HEADER ()
0334
0335      packet_length = record_size - 36
0336
0337      call LINCHK (lstlun,1)
0338
0339      10 write(lstlun,10)
0340      format(' ',:)
0341
0342      if (logmscp_type .eq. 9) then
0343
0344      mscp$$b_opcode = lib$extzv(0,7,emb(44))
0345      mscp$$b_endcode = lib$extzv(0,8,emb(44))
0346
0347      call LINCHK (lstlun,1)
0348
0349      15 write(lstlun,15) 'UNEXPECTED END MESSAGE 'MSCP$W_STATUS''
0350      format(' ',t8,a)
0351
0352      call MOV3 (%val(packet_length),emb(36),emb(38))
0353      call MSCP$B_OPCODE_DISPATCHER (lstlun,mscp$$b_endcode,..true...false.)
0354
0355      else if (logmscp_type .eq. 10) then
0356
0357      mscp$$b_opcode = lib$extzv(0,7,emb(44))
0358      mscp$$b_endcode = lib$extzv(0,8,emb(44))
0359
0360      call LINCHK (lstlun,1)
0361      write(lstlun,15) 'UNEXPECTED ATTENTION CONTROL MESSAGE'
0362
0363      call MOV3 (%val(packet_length),emb(36),emb(38))
0364
0365      call LINCHK (lstlun,1)
0366      write(lstlun,10)
0367
```



```
0368      call MSCP_FIRST_TWELVE_BYTES2 (lstlun,.true.)
0369
0370      if (packet_length .gt. 12) then
0371
0372      call LINCHK (lstlun,1)
0373      write(lstlun,10)
0374
0375      call DUMPREG (lstlun,(((packet_length - 12) + 3)/4),emb(50))
0376      endif
0377
0378      else if (logmscp_type .eq. 11) then
0379
0380      mslg$$b_format = lib$extzv(0,7,emb(44))
0381
0382      call LINCHK (lstlun,1)
0383
0384      if (
0385      1 mslg$$b_format .eq. 0
0386      1 .or.
0387      1 mslg$$b_format .eq. 1
0388      1 ) then
0389
0390      write(lstlun,15) 'NO-UNIT DATAGRAM'
0391      else
0392
0393      write(lstlun,15) 'DATAGRAM FOR NON-EXISTING "UCB"'
0394      endif
0395
0396      c
0397      c      Although the 'emblmdef' format is not part of this kind of error log
0398      c      entry it is necessary, because of the implementation chosen, to zero
0399      c      the equivalent 'emb$b_lm_type' field and re-position the "mscp" portion
0400      c      of the entry for the output routines.
0401      c
0402
0403      call MOVC3 (%val(packet_length),emb(36),emb(38))
0404
0405      emb$b_lm_type = 0
0406
0407      if (mslg$$b_format .eq. 0) then
0408      call MSLG$K_CNT_ERR (lstlun,packet_length)
0409
0410      else if (mslg$$b_format .eq. 1) then
0411      call MSLG$K_BUS_ADDR (lstlun,packet_length)
0412
0413      else if (
0414      1 mslg$$b_format .eq. 2          ! Disk transfer error
0415      1 .or.
0416      1 mslg$$b_format .eq. 5          ! Tape transfer error
0417      1 ) then
0418      call DISK_TAPE_TRANSFER_ERRORS (lstlun,packet_length)
0419
0420      else if (
0421      1 mslg$$b_format .eq. 3          ! SDI error
0422      1 .or.
0423      1 mslg$$b_format .eq. 6          ! STI error
0424      1 .or.
```

```
0425      1 mslg$$b_format .eq. 7          ! STI drive error
0426      1 .OR.
0427      1 mslg$$b_format .eq. 8          ! STI formatter error
0428      1 ) then
0429      call SDI_STI_ERRORS (lstlun,packet_length)
0430
0431      else if (mslg$$b_format .eq. 4) then ! Small disk error
0432      call MSLG$K_SML_DSK (lstlun,packet_length)
0433      else
0434
0435      call LINCHK (lstlun,1)
0436
0437      write(lstlun,10)
0438
0439      call DUMPREG (lstlun,(((packet_length) + 3)/4),emb(38))
0440      endif
0441
0442      c
0443      c      The IF-THEN-ELSE should be extended at this point to add
0444      c      additional "logmscp_type" formats.
0445      c
0446
0447      else
0448
0449      call LINCHK (lstlun,1)
0450
0451      write(lstlun,20) logmscp_type
0452      format(' ',t8,'ENTRY TYPE #',
20      1 i<compress4 (lib$extzv(0,16,logmscp_type))>,'.')
0454
0455      call LINCHK (lstlun,1)
0456
0457      write(lstlun,10)
0458
0459      call DUMPREG (lstlun,(((packet_length) + 3)/4),emb(36))
0460      endif
0461
0462      return
0463      end
```



## PROGRAM SECTIONS

Name	Bytes	Attributes
0 \$CODE	798	PIC CON REL LCL SHR EXE RD NOWRT LONG
1 \$PDATA	208	PIC CON REL LCL SHR NOEXE RD NOWRT LONG
2 \$LOCAL	260	PIC CON REL LCL NOSHR NOEXE RD WRT LONG
3 EMB	512	PIC OVR REL GBL SHR NOEXE RD WRT LONG
4 SYECOM	44	PIC OVR REL GBL SHR NOEXE RD WRT LONG
Total Space Allocated	1822	

## ENTRY POINTS

Address	Type	Name
0-00000000		LOGMSCP

## VARIABLES

Address	Type	Name	Address	Type	Name
4-00000012	L*1	CP_11750	4-00000011	L*1	CP_11780
4-00000013	L*1	CP_11722	4-00000014	L*4	CRYPTK_FLAG
4-0000000D	I*4	DEV_CHAR	3-00000010	L*1	EMBSB_LM_CLASS
3-00000014	L*1	EMBSB_LM_NAMLNG	3-00000011	L*1	EMBSB_LM_TYPE
3-00000000	I*4	EMBSL_HD_SID	3-00000015	CHAR	EMBST_LM_NAME
3-00000004	I*2	EMBSW_HD_ENTRY	3-0000000E	I*2	EMBSW_HD_ERRSEQ
3-00000024	I*2	EMBSW_LM_MSGTYP	3-00000012	I*2	EMBSW_LM_UNIT
4-0000001E	L*1	END_VALUE	4-0000001D	L*1	EOF_FLAG
4-00000004	L*4	FORMS	4-0000000C	L*1	LINES
3-00000010	I*2	LOGMSCP_TYPE	4-00000027	I*4	LSTLUN
AP-00000004a	L*1	LUN	4-0000001F	I*4	MAILBOX_CHANNEL
2-00000004	I*4	MSCP\$SB_ENDCODE	2-00000000	I*4	MSCP\$SB_OPCODE
2-00000008	I*4	MSLG\$SB_FORMAT	4-0000002B	CHAR	OPTIONS
2-0000000C	I*4	PACKET_LENGTH	4-00000008	L*4	PRINTER
4-00000000	I*4	RECCNT	AP-00000008a	I*4	RECORD_LENGTH
4-00000023	I*4	RECORD_SIZE	4-00000019	L*1	VALID_CLASS
4-0000001A	L*1	VALID_CPU	4-0000001B	L*1	VALID_ENTRY
4-0000001C	L*1	VALID_TYPE	4-00000018	L*1	VOLUME_OUTPUT

## ARRAYS

Address	Type	Name	Bytes	Dimensions
3-00000000	L*1	EMB	512	(0:511)
3-00000026	L*1	EMBSB_LM_MSGTXT	460	(460)
3-00000006	I*4	EMBSQ_HD_TIME	8	(2)
3-00000024	L*1	MSCP_PACKET	476	(476)

Page 6

Address	Label	Address	Label	Address	Label
1-000000A7	10'	1-000000AC	15'	1-000000B3	20'

Type	Name	Type	Name	Type	Name
I*4	COMPRESS4		DISK_TAPE_TRANSFER_ERRORS		DUMPREG
	FRCTOF		HEADER	I*4	LIB\$EXTZV
	LINCHK		LOGGER		LOGMSCP_HEADER
	MOVC3		MSCP\$B_OPCODE_DISPATCHER		MSCP_FIRST_TWELVE_BYTES2
	MSLG\$K_BUS_ADDR		MSLG\$K_CNT_ERR		MSLG\$K_SML_DSK
	SDI_STI_ERRORS				

[illegible]



```

0001
0002
0003
0004
0005      Subroutine LOGMSCP_HEADER ()
0006
0007
0008      include      'src$:msghdr.for/nolist'
0009      include      'src$:syecom.for/nolist'
0010
0011
0012      byte          class_driver_array(4)
0013      integer*4      class_driver
0014      character*4     class_driver_string
0015      equivalence    (emb(18),class_driver_array(1),class_driver_string,
0016      1 class_driver)
0017
0018      integer*4      cddb$q_cntrlid_array(2)
0019      integer*4      cddb$q_cntrlid1
0020      equivalence    (cddb$q_cntrlid_array(1),cddb$q_cntrlid1)
0021      integer*4      cddb$q_cntrlid2
0022      equivalence    (cddb$q_cntrlid_array(2),cddb$q_cntrlid2)
0023      equivalence    (emb(22),cddb$q_cntrlid_array(1))
0024
0025      byte          cddb$b_systemid_array(6)
0026      integer*4      cddb$b_systemid1
0027      equivalence    (cddb$b_systemid_array(1),cddb$b_systemid1)
0028      integer*2      cddb$b_systemid2
0029      equivalence    (cddb$b_systemid_array(5),cddb$b_systemid2)
0030      equivalence    (emb(30),cddb$b_systemid_array(1))
0031
0032      byte          fao_array(4)
0033      character*4     fao_string
0034      equivalence    (fao_array(1),fao_string)
0035
0036      external      sys$fao
0037      logical*4      sys$fao
0038
0039
0040      call LINCHK (lstlun,2)
0041
0042      write(lstlun,10) 'CLASS DRIVER',class_driver
0043      format(/' ',t8,a,t24,z8.8)
0044
0045      if (sys$fao('!AF',,fao_string,xval(4),class_driver_array(1))) then
0046
0047      call LINCHK (lstlun,1)
0048
0049      write(lstlun,15) fao_string
0050      format(' ',t40,'/',a,'/')
0051      endif
0052
0053      call LINCHK (lstlun,2)
0054
0055      write(lstlun,20) cddb$q_cntrlid1,cddb$q_cntrlid2
0056      format(' ',t8,'CDDB$Q_CNTRLID',t24,z8.8,/,
0057      1 t24,z8.8)

```

```

0243
0244      call CDDBSQ_CNTRLID (lstlun,cddb$q_cntrlrid1,cddb$q_cntrlrid2)
0245
0246      call LINCHK (lstlun,2)
0247
0248      write(lstlun,25) cddb$b_systemid1,cddb$b_systemid2
0249 25      format(/' ',t8,'CDDBSB_SYSTEMID',t24,z8.8,/,
0250          1 t28,z4.4)
0251
0252      return
0253
0254      end

```

## PROGRAM SECTIONS

Name	Bytes	Attributes
0 \$CODE	242	PIC CON REL LCL SHR EXE RD NOWRT LONG
1 \$PDATA	118	PIC CON REL LCL SHR NOEXE RD NOWRT LONG
2 \$LOCAL	96	PIC CON REL LCL NOSHR NOEXE RD WRT LONG
3 EMB	512	PIC OVR REL GBL SHR NOEXE RD WRT LONG
4 SYECOM	44	PIC OVR REL GBL SHR NOEXE RD WRT LONG
Total Space Allocated	1012	

## ENTRY POINTS

Address	Type	Name
0-00000000		LOGMSCP_HEADER

## VARIABLES

Address	Type	Name	Address	Type	Name
3-0000001E	I*4	CDDBSB_SYSTEMID1	3-00000022	I*2	CDDBSB_SYSTEMID2
2-00000004	R*4	CDDBSQ_CNTRLID1	3-00000016	I*4	CDDBSQ_CNTRLID1
3-0000001A	I*4	CDDBSQ_CNTRLID2	3-00000012	I*4	CLASS_DRIVER
3-00000012	CHAR	CLASS_DRIVER_STRING	4-00000012	L*1	CP_11750
4-00000011	L*1	CP_11780	4-00000013	L*1	CP_11722
4-00000014	L*4	CRYPTK_FLAG	4-0000000D	I*4	DEV_CHAR
3-00000000	I*4	EMBSL_HD_SID	3-00000004	I*2	EMBSW_HD_ENTRY
3-0000000E	I*2	EMBSW_HD_ERRSEQ	4-0000001E	L*1	END_VALUE
4-0000001D	L*1	EOF_FLAG	2-00000000	CHAR	FAO_STRING
4-00000004	L*4	FORMS	4-0000000C	L*1	LINES
4-00000027	I*4	LSTLUN	4-0000001F	I*4	MAILBOX_CHANNEL
4-0000002B	CHAR	OPTIONS	4-00000008	L*4	PRINTER
4-00000000	I*4	RECCNT	4-00000023	I*4	RECORD_SIZE
4-00000019	L*1	VALID_CLASS	4-0000001A	L*1	VALID_CPU
4-0000001B	L*1	VALID_ENTRY	4-0000001C	L*1	VALID_TYPE
4-00000018	L*1	VOLUME_OUTPUT			



## ARRAYS

Address	Type	Name	Bytes	Dimensions
3-0000001E	L*1	CDDBSB_SYSTEMID_ARRAY	6	(6)
3-00000016	I*4	CDDBSQ_CNTRLID_ARRAY	8	(2)
3-00000012	L*1	CLASS_DRIVER_ARRAY	4	(4)
3-00000000	L*1	EMB	512	(0:511)
3-00000006	I*4	EMBSQ_HD_TIME	8	(2)
2-00000000	L*1	FAO_ARRAY	4	(4)

## LABELS

Address	Label	Address	Label	Address	Label	Address	Label
1-00000018	10'	1-00000025	15'	1-00000032	20'	1-00000053	25'

## FUNCTIONS AND SUBROUTINES REFERENCED

Type	Name	Type	Name	Type	Name
	CDDBSQ_CNTRLID		LINCHK		L*4 SYSS\$FA0

## COMMAND QUALIFIERS

FORTRAN /LIS=LIS\$:LOGMSCP/OBJ=OBJ\$:LOGMSCP MSRC\$:LOGMSCP

/CHECK=(NOBOUNDS,OVERFLOW,NOUNDERFLOW)

/DEBUG=(NOSYMBOLS,TRACEBACK)

/STANDARD=(NOSYNTAX,NOSOURCE\_FORM)

/SHOW=(NOPREPROCESSOR,NOINCLUDE,MAP)

/F77 /NOG\_FLOATING /I4 /OPTIMIZE /WARNINGS /NOD\_LINES /NOCROSS\_REFERENCE /NOMACHINE\_CODE /CONTINUATIONS=19

## COMPILATION STATISTICS

Run Time:	4.86 seconds
Elapsed Time:	14.09 seconds
Page Faults:	187
Dynamic Memory:	183 pages



